

Práctica 1:
Paralelización de un algoritmo de compresión de imágenes volumétricas con Threads.

Laboratorio de Arquitecturas Avanzadas de Computadores.





Introducción

- En esta práctica se va a realizar una paralelización de un algoritmo de compresión de imágenes 3D usando las librerías pthreads, usando una implementación de la transformada wavelet y un algoritmo de compresión RLE.





1 Objetivos

- Poner en práctica los conocimientos adquiridos en los temas 1 y 2 de la asignatura, sobre un multiprocesador Silicon Graphics Origin 2000 con 8 procesadores.







1 Objetivos

- Este objetivo principal está compuesto de los siguientes objetivos:
  - Estudiar el diseño y la implementación secuencial del algoritmo de compresión de imágenes volumétricas.
  - Estudiar la viabilidad del proyecto paralelo.
  - Estudiar y diseñar la parte paralela, división en fases, problemas de cada una.
  - Implementar el diseño paralelo usando threads sobre una arquitectura de memoria compartida Origin 2000.
  - Realizar pruebas de rendimiento sobre las implementaciones secuencial y paralela y sacar conclusiones.





2 La transformada wavelet

- La transformada discreta de Wavelet (DWT) es una función ortogonal la cual puede ser aplicada a un grupo finito de datos. Este grupo finito de datos puede ser modelado como una función f(t) en términos de contracciones y dilataciones de una función de escalado (t) y una función Wavelet (t). La expresión general de la función f(t) queda de la siguiente forma:

$$f(t)=\sum_{k} c_k^J 2^{-\frac{J}{2}} \phi\left(2^{-J} t-k\right)+\sum_{j=-\infty}^{J} \sum_{k} d_k^j 2^{-\frac{j}{2}} \varphi\left(2^{-j} t-k\right)$$







2 La transformada wavelet

- En esta expresión se distinguen dos términos:
  - Una aproximación a f(t) en una escala o resolución "J". A estos términos c_k^j se les denomina coeficientes de escalado (scaling coefficients).

$$c_k^J = 2^{-\frac{J}{2}} \int f(t)\phi\left(2^{-J}t - k\right)$$

  - Un refinamiento de la función en una escala j < J. En él se encuentran los coeficientes d_k^j, conocidos como coeficientes Wavelet (Wavelet coeffiecients).

$$d_k^j = 2^{-\frac{j}{2}} \int f(t)\varphi\left(2^{-j}t - k\right)$$







2 La transformada wavelet

- Discretizando las funciones anteriores obtenemos las siguientes:

$$\phi\left(\frac{t}{2}\right)=\sqrt{2}\sum_{n} h[n]\phi(t-n)$$

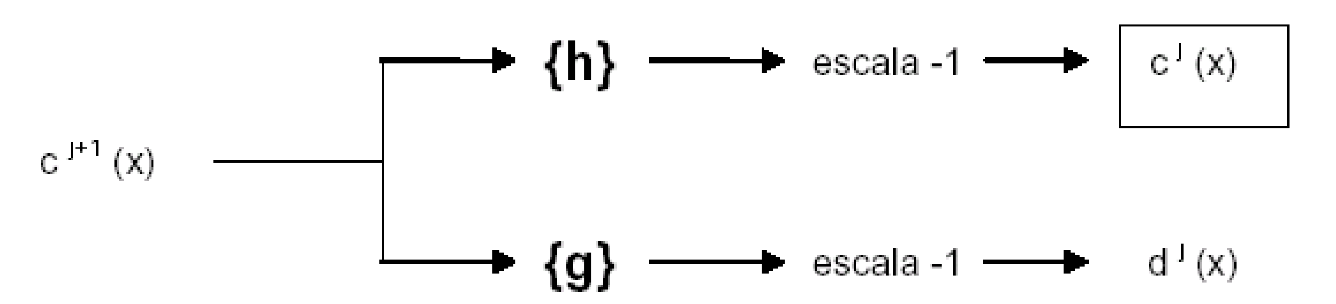
$$\varphi\left(\frac{t}{2}\right)=\sqrt{2}\sum_{n} g[n]\varphi(t-n)$$

- Donde h[n] y g[n] corresponden a los filtros paso-alto y paso-bajo respectivamente.



2 La transformada wavelet

- Estas funciones reciben un número finito de N datos discretos ($c^{j+1}(x)$), donde N es potencia de dos, estos datos se pasan a través de los filtros paso-alto y paso-bajo, descomponiéndose en dos subbandas ($d^{j}(x)$ y $c^{j}(x)$). Cada una de estas subbandas contendrá N/2 datos.







2 La transformada wavelet

- Los coeficientes c^j (x) conforman la subbanda de frecuencias bajas, representada comúnmente con L (Low), del mismo modo, los coeficientes d^j (x) corresponden a la subbanda de frecuencias altas H (High).
- El algoritmo puede aplicarse varias veces, de manera que la señal ckj+1 (x) se descompondrá sucesivamente en escalas j, j-1, j-2…
- De esta forma se aumenta el número de subbandas en las que se divide la imagen.



3 Tratamiento de imágenes 3D mediante Wavelet

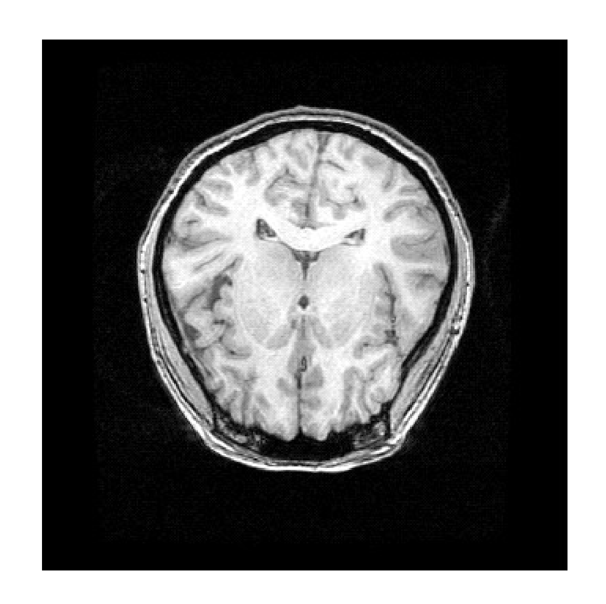


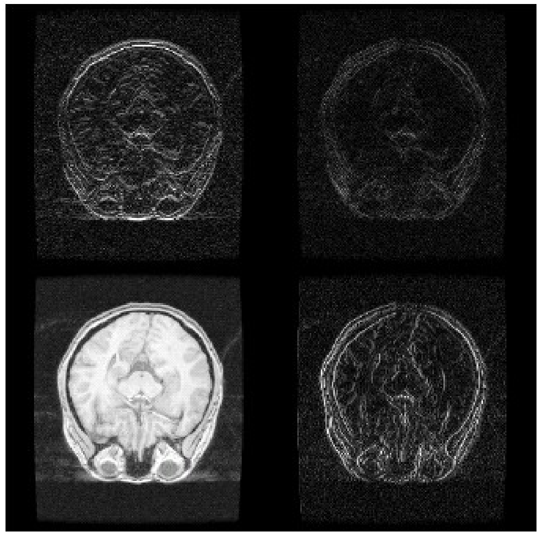
- Los Wavelet proporcionan un método de análisis ya que descompone la imagen en coeficientes de aproximación (coeficientes de escalado), generados por el filtro paso-bajo y coeficientes de detalle (coeficientes Wavelet), generados por el filtro paso-alto.
- Al aplicarlo a una imagen 3D ,obtendremos otra imagen con el mismo número de elementos descompuesta en subbandas. Para ello aplicaremos la descomposición unidimensional sobre el eje x, al resultado obtenido se le aplicará la descomposición unidimensional sobre el eje y, y finalmente sobre el eje z.

3 Tratamiento de imágenes 3D mediante Wavelet

Universidad Rey Juan Carlos

- El resultado de aplicar la transformada Wavelet sobre cada uno de los ejes de la imagen 3D hace que ésta quede dividida en ocho subcubos, cada uno correspondiente a una subbanda:



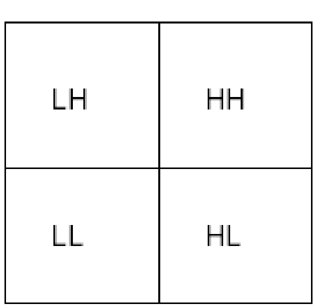


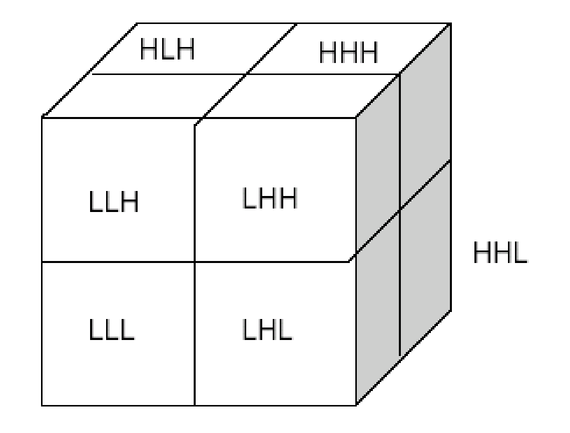


3 Tratamiento de imágenes 3D mediante Wavelet



- Subbanda LLL: contiene las frecuencias más bajas de la imagen, correspondiente a los coeficientes de escalado, estos son los coeficientes que más información aportan.
- Subbandas LLH, LHL, LHH, HLL, HLH, HHL, HHH: contienen el resto de frecuencias de la imagen, correspondientes a los coeficientes Wavelets.



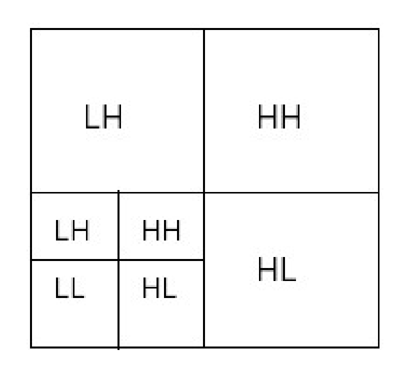


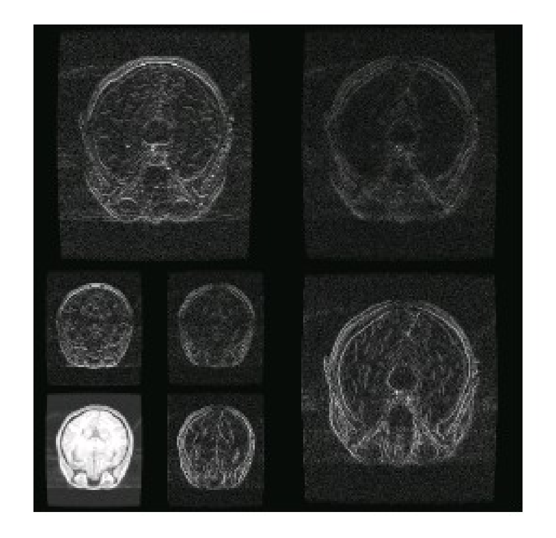




3 Tratamiento de imágenes 3D mediante Wavelet

- El algoritmo puede aplicarse varias veces, aunque sólo en la subbanda de coeficientes de escalado (LL), esquematizado en la figura siguiente sobre una imagen 2D:









3 Tratamiento de imágenes 3D mediante Wavelet

- Este método de transformación proporciona una separación en subbandas,pero el resultado contiene exactamente los mismos elementos que la señal discreta inicial.
- Una vez se tiene la imagen descompuesta, se le puede aplicar algún algoritmo de compresión. Para ello se usará el algoritmo RLE (Run Length Encoding).





4. Desarrollo de la práctica



- Pasos a seguir para realizar la práctica:
  - Realizar una versión secuencial del algortimo.
  - Estudio de viabilidad para una versión paralela.
    - Ley de Amdalh.
    - Gprof.
  - Realizar una versión paralela del algoritmo.
    - Pthreads
  - Pruebas de rendimiento.
  - Conclusiones.







4. Desarrollo de la práctica

Gprof:

- Nos da información sobre el tiempo de ejecución de un determinado método, las veces que es invocado, etc...
- Para poder usarlo, hay que habilitar la opción “-pg” del gcc, de forma que introduzca instrucciones de “profiling” en el código.





4. Desarrollo de la práctica

Modo de empleo:

- Ejecutar el programa compilado con la opción "–pg". Esto generará el fichero gmon.out, donde guardará información necesaria para gprof.
- Ejecutar "gprof _nombre_programa_ > fich_resultado" para que interprete los datos guardados en gmon.out, y genere la información.

```
Flat profile:

Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls  us/call  us/call  name
 94.27      6.09     6.09   689640     8.83     8.83  inner
  3.56      6.32     0.23      112  2053.57  2053.57  initialize
  1.08      6.39     0.07      112   625.00 55000.00  multiply
  1.08      6.46     0.07      112   625.00   625.00  writeMat
```





4. Desarrollo de la práctica

- Tanto para la implementación secuencial como para la paralela se proporcionarán al alumno una serie de librerías con las funciones necesarias para realizar dicha implementación. Estos módulos son los siguientes:

float16.c/h: Implementa un tipo nuevo de 16 bits en coma fija: 1 bit de signo, 12 bits parte entera, 3 bits parte fraccionaria

mat_util.c/.h: Conjunto de funciones matemáticas, necesarias para la implementación de la transformada 3D.

wavelet.c/.h: Implementación de la transformada wavelet de Haar 3D.

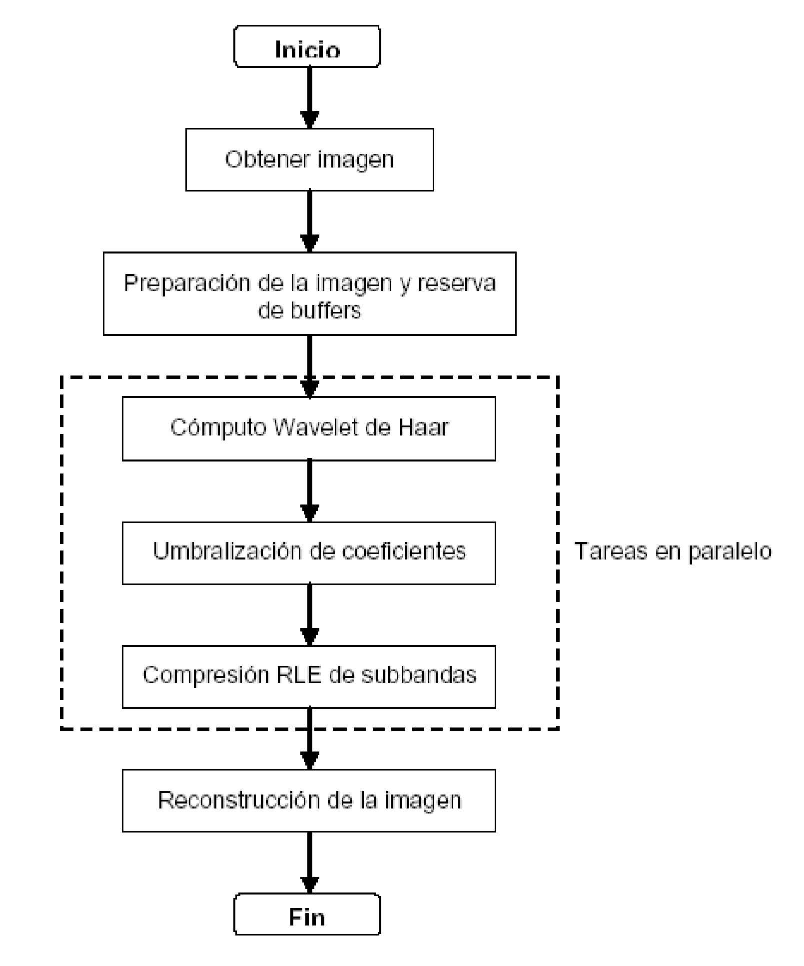
compress.c, compress.h:Implementación del algoritmo de compresión RLE.

DAC



4. Desarrollo de la práctica

- A continuación se muestra el diagrama de flujo del programa principal:





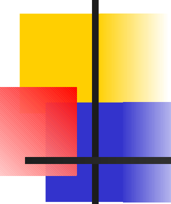


4. Desarrollo de la práctica

- 1. La lectura de la imagen se realiza desde el fichero de imagen de entrada proporcionado. La imagen se guarda en formato float16, y es un cubo de m filas por n columnas y k imágenes de profundidad.
- 2. Ejecutar la función de que realiza la transformada wavelet de Haar 3D, que se proporciona en el módulo wavelet.c.
- 3. Realizar la umbralización y compresión mediante el algoritmo RLE utilizando la función proporcionada en el módulo compress.
- 4. Escribir la imagen transformada en un fichero de salida. Para ello se debe escribir en primer lugar la cabecera, con la función proporciona en el módulo compress.c y a continuación escribir los datos.







5. Parámetros de entrada.

- El programa toma como parámetros de entrada los siguientes:
  - Nombre del fichero de imagen
  - Número de columnas de la imagen de entrada.
  - Número de filas de la imagen de entrada.
  - Número de rodajas que componen el volumen.
  - Umbral para el proceso de compresión.
- Para el fichero de entrada que se proporciona como ejemplo los valores deben ser:



Nombre_programa im.img 512 512 128 1

6. Entrega y evaluación de la práctica



- La práctica se podrá entregar hasta el día 5 de Diciembre de 2006.
- La entrega se realizará en horas de clase en el laboratorio 108.
- La entrega constará de dos partes:
  - Examen oral de la práctica (Presentación de la práctica, compilar y ejecutar).
  - Una memoria explicativa de la práctica.
- Evaluación:
  - El examen oral en el laboratorio.
  - Los tiempos de respuesta obtenidos.
  - La memoria explicativa.

